

Amendments to the Claims:

This listing of claims replaces all prior versions and listings of claims in the application:

Listing of Claims:

1. (Currently amended) A computer program product residing on a computer readable storage medium comprising instructions, including a context branch instruction that, when executed causes a data processing apparatus to:

cause an instruction stream to branch to another instruction of the instruction stream associated with a label specified by the context branch instruction based on an evaluation of whether ~~or not~~ a current context number matches a context number specified by the context branch instruction.

2. (Previously presented) The computer program product of claim 1 wherein the branch instruction has one of the following formats:

br=ctx[ctx#, label#], optional\_token; and

br!=ctx[ctx#, label#], optional\_token.

3. (Previously presented) The computer program product of claim 2 wherein the label# is a symbolic label corresponding to an address of the other instruction, and wherein ctx# is the context number.

4. (Previously presented) The computer program product of claim 3 wherein the specified context number has valid values of 0, 1, 2, or 3.

5. (Previously presented) The computer program product of claim 1 wherein the branch instruction has an optional token.

6. (Previously presented) The computer program product of claim 5 wherein the context branch instruction has an optional token that causes a processor to execute a number of instructions corresponding to the value of the optional token following the context branch instruction before performing the branch operation.

7. (Previously presented) A method of operating a processor comprising:  
evaluating a context number of an executing context to determine whether the context number of the executing context matches a context number specified by a context branch instruction; and

branching to a specified instruction in accordance with evaluating the context number of the executing context.

8. (Original) The method of claim 7 wherein branching further comprises:  
branching if the executing context number matches the specified context number.

9. (Original) The method of claim 7 wherein the context number has valid values of 0, 1, 2, or 3.

10. (Previously presented) A processor that can execute multiple contexts and that comprises:

a register stack;

a program counter for each executing context;

an arithmetic logic unit coupled to the register stack and a program control store that stores a context branch instruction that causes the processor to:

evaluate a context number of an executing context to determine whether the context number of the executing context matches a context number specified by the branch instruction; and

branch to a specified instruction in accordance with evaluating the context number of the executing context.

11. (Original) The processor of claim 10 wherein a branch occurs if the executing context number matches the specified context number.

12. (Original) The processor of claim 10 wherein the context number has valid values of 0, 1, 2, or 3.

13. (Currently amended) A computer program product residing on a computer readable storage ~~medium~~ medium, for causing a processor that executes multiple contexts to perform a ~~function~~ function, comprises instructions causing the processor to:

evaluate a context number of an executing context to determine whether the context number of the executing context matches a context number specified by a branch instruction; and

branch to a specified instruction in accordance with evaluating the context number of the executing context.

14. (Original) The product of claim 13 wherein a branch occurs if the executing context number matches the specified context number.

15. (Original) The product of claim 13 wherein the context number has valid values of 0, 1, 2, or 3.